IFW 2138

TRANSMITTAL LETTER
(General - Patent Pending)

Docket No.
L9289.00121

In Re Application Of: **Kuniyuki KAJITA**

Application No.	Filing Date	Examiner	Customer No.	Group Art Unit	Confirmation No.
09/701,433	**November 29, 2000**	**P. Chung**	**52989**	**2138**	**9782**

Title: **RADIO COMMUNICATION APPARATUS AND CODING PROESSING METHOD**

COMMISSIONER FOR PATENTS:

Transmitted herewith is:

A Reply Brief (in triplicate) in response to the Examiner's Answer of November 13, 2006 with copies of the Hagenauer document (in triplicate)

in the above identified application.

☒ No additional fee is required.

☐ A check in the amount of is attached.

☒ The Director is hereby authorized to charge and credit Deposit Account No. **19-4375** as described below.

- ☐ Charge the amount of
- ☒ Credit any overpayment.
- ☒ Charge any additional fee required.

☐ Payment by credit card. Form PTO-2038 is attached.
WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Signature

Dated: **January 16, 2007**

James E. Ledbetter, Esq.
Registration No. 28,732
Stevens, Davis, Miller & Mosher, LLP
1615 L Street, N.W., Suite 850
Washington, DC 20036
Tel: 202-785-0100
Fax: 202-408-5200

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to the "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)] on

_____________________.
(Date)

Signature of Person Mailing Correspondence

Typed or Printed Name of Person Mailing Correspondence

cc:

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of

Inventor(s):	Kuniyuki KAJITA	Art Unit 2133
Appln. No.:	09/701,433	Exr. P. M. Chung
Filed:	November 29, 2000	
For:	RADIO COMMUNICATION APPARATUS AND CODING PROCESSING METHOD	

REPLY BRIEF UNDER 37 CFR 41.41

Assistant Commissioner of Patents
Washington, DC 20231

Dear Sir:

This Reply Brief is submitted in order to rebut various arguments in the Examiner's Answer of November 13, 2006 and to address the new grounds of rejection as set forth in 37 CFR 41.37(c)(1)(vii), following the other requirements of a brief as set forth in 37 CFR 41.37(c). The Appellant hereby requests, by filing this Reply Brief, that the appeal be maintained as set forth in 37 CFR 41.39(b)(2).

With respect to the provisions of 37 CFR 41.37 referenced in 37 CFR 41.39(b)(2), the Appellant notes that the Real Party in Interest, Related Appeals and Interferences, Status of Claims, Status of Amendments, Summary of Claimed Subject Matter and Grouping of Claims are as set forth in the corrected Brief filed August 11, 2006. An Evidence Appendix identifies the Hagenauer article ("Rate-Compatible Punctured Convolutional Codes (RCPC Codes) and their Applications") discussed in the Main Brief filed August 11, 2006.

The grounds of rejection on appeal were revised in the Examiner's Answer. In light of this, as required by 37 CFR 41.37(c)(1)(vii), the following is a concise statement of the current grounds of rejection presented for review herein:

A. Pending Grounds of Rejection to Be Reviewed on Appeal

(1) Claims 11-13, 19, 21-25, 31, 33, 34, 36, 38, 41 and 43 stand rejected under 35 USC 103(a) as unpatentable over Chen et al. (USPN 6,199,186) in view of Frenger et al. ("Rate Matching in Multi-Channel Systems Using RCPC-Codes") and further in view of Akihiko Watanabe et al. (Study on Wideband CDMA System in Burst Error Environment").

(2) Claim 20, 32, 35, 37, 39, 40, 42 and 44 stand rejected under 35 USC 103(a) as unpatentable over Chen et al. (USPN 6,199,186) in view of Frenger et al. ("Rate Matching in Multi-Channel Systems Using RCPC-Codes") and further in view of Akihiko Watanabe et al. (Study on Wideband CDMA System in Burst Error Environment"), as applied to claims 11 and 19, and further in view of the Applicants' admitted prior art (Figs. 1A, 1B and 2).

B. Argument

1. Underlying Concept of the Invention

First of all, the Appellant wishes to explain the functions of each of error correction coding, interleaving and rate matching.

Error correction coding, such as convolutional coding and turbo coding, adds redundancy to enable correction of the errors in received data.

Interleaving involves distributing (deconcentrating) the error resistibility by changin the order of bits; it is used for improving resistibility to burst errors of data.

Rate matching involves adapting to various data rates by repeating or puncturing a part of bits. It is used for adjusting the data length of data, e.g., repeating bits of data which are shorter than a requested length or puncturing bits of data which are longer than the requested length. Also, as an adventitious function, rate matching is operable to equalize the resistibility to the error of the whole data, as discussed in detail hereinafter, by repeating or puncturing of a part of bits in the whole data, preferably at a regular interval.

Accordingly, the functions of the above operations of error correction encoding, interleaving and rate matching are completely different.

The present inventor discovered that the prior art technique, shown in application Figs. 1A and 1B, suffers from burst error problems because the operation of interleaving following repeating (rate matching) causes complexity of interleaving (it needs to be adapted to various data lengths) and unequalizing (centralizing) of the repeated or punctured bits.

Therefore, to avoid these problems, the present inventor conceived the idea of a coding process of performing the error correction coding, then the interleaving, and then the rate matching. This provides advantages of adaptation to various data rates by repeating or puncturing a part of bits (claims 11 and 31), selecting alternatively between repeating and puncturing (claim 11), and interleaving immediately after error correction coding (interleaving need not be adapted to various data rates (claims 11 and 31). Another advantage involves improving the resistibility to burst error (equalizing the error resistibility) by interleaving followed by rate matching (claims 11 and 31) avoiding unequalizing caused by interleaving following repeating or puncturing a part of bits (claim 13). That is, the present claimed invention provides a coding process which has advantages of both <u>improving</u> the error resistibility and <u>equalizing</u> the error resistibility and of achieving high resistibility to burst error. The improving and equalizing of the resistibility provides the further advantage of improving the performance of the error correcting at the receiver (the decoding process).

Also, in the present claimed invention, another advantage is obtained in that the interleaving does not need to be adapted to various data rates. That is, in the present invention, data input into the interleaving has a predetermined rate because the interleaving is performed immediately after the error correction coding. Therefore, the interleaving does not need to be adapted to many data rates accompanied with the rate matching.

2. Rejection of Claims 11-13, 19, 21-25, 31, 33, 34, 36, 38, 41 and 43 under 35 USC 103(a)

The Examiner's Answer discusses this rejection in two sections identified below.

A. Claims 11, 13, 21, 22, 36 and 38

With respect to claims 11, 13, 21, 22, 36 and 38, the Examiner's Answer states that Chen et al. disclose a transmission system comprising a coder that performs error correction coding of input data including a plurality of bits, and an interleaver that performs interleaving of the bits coded by the coder, but lacks any disclosure of a rate matcher that comprises a repeater and a puncturer wherein the rate matcher alternatively selects between employing the repeater to repeat a part of the bits interleaved by the interleaver and employing the puncturer to puncture a part of the bits interleaved by the interleaver. The Examiner's Answer states that Frenger et al. disclose a rate matcher that comprises a repeater and a puncturer, but lacks any disclosure of whether the rate matching is performed before or after interleaving.

Page 4 of the Examiner's answer states that "However, it would have been obvious ... to set the rate matcher to either the rate matching followed by interleaving is performed after convolutional code after coding *to reduce burst error* or interleaving followed by rate matching is performed after convolutional coding *to reduce burst error.*" (Emphasis added). (The advisory action stated the motivation for the combination was "to reduce latency." The Appellant pointed out in the main Brief that this statement is not comprehensible. The statement has been dropped in the Examiner's Answer). The Examiner's Answer further states that "Therefore, it would have been obvious ... to combine the rate matching of Frenger into the interleaving performed after coding of Chen to rate match *for reducing burst error*. Thus, the combination of Chen et al and Frenger et al would be performed the interleaving before the rate matching." (Emphasis added). The Examiner's Answer further states that "In addition to the missing part of Chen et al and Frenger et al, Akihiko et al disclose a rate matcher that is performed after interleaving (See pp. 325, "Figure 1, Burst Error Reduction methods"). Therefore, it would have been obvious ... to incorporate a rate matcher that is performed after interleaving as taught by Akihiko et al into the invention of Chen et al and

Frenger et al to rate match *for reducing burst errors*." (Emphasis added). With respect to claim 19, the Examiner's Answer states that this method claim is rejected under a similar rationale as set forth against system claim 11.

The Main Brief filed August 11, 2006 fully identified and discussed the deficiencies of Chen et al. and Frenger et al., and this Reply Brief incorporates that discussion by reference.

The Examiner's answer alleges that it would have been obvious to combine the rate matching of Frenger et al. into the interleaving performed after coding of Chen et al. to rate match for reducing burst error. However, the Appellant notes that there is no motivation to combine the rate-matching of Frenger into Chen because Chen et al. do not teach or suggest adapting to various data rates, and Chen et al. and Frenger et al. do not teach or suggest increasing resistibility to burst error. Reducing burst error cannot be a motivation to combine these references because rate-matching is used for adapting the data rate not for reducing burst error. The rate-matching of Frenger is performed within a coder (RCPC) or repeater concatenated with a convolutional coder. The rate-matching is performed within or immediately after error correction coding if both are combined. The repeater of Frenger is used as comparison with the RCPC. The repeater of Frenger is not alternatively selected from rate-matching including both puncturing and repeating.

Regarding the motivation alleged in the Examiner's Answer, the Appellant submits that the motivation of "burst error reduction" would not have led a person skilled in the art at the time the present invention was made to incorporate the rate matching of Frenger et al. into Chen et al. This is because the main function of rate matching is to adjust the data length, not to reduce burst error. The reduction of burst error is achieved by the interleaving, and the interleaving is disclosed in Chen et al. without the need to refer to Frenger et al. Thus, the reduction of burst error is neither a motivation nor a rationale for combining the rate matching of Frenger et al. into Chen et al. It is submitted that the use of this rationale or motivation in the Examiner's Answer is unwarranted and is based on improper hindsight.

With respect to Akihiko Watanabe et al., the Appellant notes that, in this reference, the rate matching is performed for adjusting the data length to a frame length, and there are a variety of frame lengths depending on the service and data to be transmitted. Therefore, the rate-matching needs to be adapted to repeat or puncture a part of the bits of a bit sequence as well as to repeat every bit of the bit sequence. In the case that every bit is repeated, the unequalizing of the resistibility to the error does not arise due to the interleaving because there is no difference in the resistibility to the error for each bit (because there are no non-repeated bits). On the other hand, in the case that only a part of the bits is repeated or punctured, the unequalizing of the resistibility to the error arises due to the interleaver because there is a difference in the resistibility to the error between bits (because some bits are repeated and others are not; that is, the repeated bits have higher resistance). In other words, a repeated bit has high resistibility to error compared with a non-repeated bit, and a non-punctured bit has high resistibility to the error compared with a punctured bit. Thus, it could arise that bits which have low resistibility to error are gathered in a portion of the bit sequence (unequalizing of the resistibility to the error) due to the interleaving.

The present inventor recognized the problem of the unequalizing of the resistibility of the error in the case that a part of the bits is repeated or punctured as explicitly recited in independent claims 11, 19, 23 and 31.

On the other hand, in Watanabe, the repetition repeats every bit as described in lines 12-14 in the left column of page 325. This is also clear from Fig. 1 which shows 64 Kbps as output from the repetition of input bits as 32 Kbps. That is, Watanabe repeats every bit and never a part of the bit sequence. Therefore, in Watanabe, there is no recognition of the problem to be solved by the claimed invention, so that there is no advantage of performing the interleaving before rate matching. Actually, Akihiko Watanabe's Figures show that the repetition followed by interleaving (Rep-Int method) is superior to the interleaving followed by the repetition (Int-Rep method).

Moreover, Akihiko Watanabe et al. do not disclose the rate matching defined in the Appellant's claims which comprise alternatively repeating and puncturing.

Further, as recited in dependent claims 13 and 25, the repetition and the puncturing of the part of the bits are performed at regular intervals in order to adequately equalize the resistibility to the error for the bit sequence in the case that a part of the bits is repeated or punctured. None of references disclose this feature, and thus these claims provide a further basis for allowability.

In summary, there is no motivation to combine the rate matcher of Akihiko Watanabe et al. into the systems of Chen et al. and Frenger et al. Chen et al. do not teach or suggest the adapting the various data rates. Chen et al. and Frenger et al. do not teach or suggest the increasing the resistibility to burst error. There is no motivation to replace the rate-matcher of Frenger et al. with the rate-matcher of Watanabe et al. Reducing burst error cannot be a motivation to combine these references. The rate-matcing is used for adapting the data rate not for reducing burst error. The rate-matching of Frenger et al. is performed within a coder (RCPC) or repeater concatenated with a convolutional coder. The rate-matching is performed within or immediately after error correction coding if combined. The rate-matching of Akihiko Watanabe et al. can not be adapted to various data rates. Every bits not a part of bits is repeated twice (adapted only to double the data length). Akihiko Watanabe et al. do not teach a problem of unequalizing (concentrating) of repeated bits. There is no problem of unequalizing of the repeated bits because every bits is repeated. Akihiko Watanabe et al. teach that the interleaving after repeating has good performance for the burst error compared with the interleaving before repeating. If a person skilled in the art intends to reduce the burst error, the interleaving would be performed after rate-matching as taught by Akihiko Watanabe et al. The difference of the effect from the claimed invention would be due to whether every bits or a part of bits is repeated.

Thus, it is submitted that claims 11 and 19 and all claims dependent therefrom are allowable for the reasons given above and in the main Brief.

B. Claims 23, 25, 33, 34, 41 and 42

With respect to claims 23, 25, 33, 34, 41 and 42, the Examiner's Answer states that Chen et al. disclose a transmission system comprising a coder that performs error correction coding of input data including a plurality of bits, and an interleaver that performs interleaving of the bits coded by the coder, but lacks any disclosure of a rate matcher that repeats a part of the bits interleaved by the interleaver. The Examiner's Answer states that Frenger et al. disclose a rate matcher including a repeater, but lacks any disclosure of whether the rate matching is performed before or after interleaving.

Page 6 of the Examiner's answer states that "However, it would have been obvious ... to set the rate matcher to either the rate matching followed by interleaving is performed after convolutional code after coding *to reduce burst error* or interleaving followed by rate matching is performed after convolutional coding *to reduce burst error.*" (Emphasis added). (As noted previously, the advisory action stated the motivation for the combination was "to reduce latency." The Appellant pointed out in the main Brief that this statement is not comprehensible. the statement has been dropped in the Examiner's Answer). The Examiner's Answer further states that "Therefore, it would have been obvious ... to incorporate the rate matching of Frenger into the interleaving performed after coding of

Chen to rate match *for reducing burst error*. Thus, the combination of Chen and Frenger would be performed the interleaving before the rate matcher." (Emphasis added). The Examiner's Answer further states that "In addition for the missing part of Chen et al and Frenger et al, Akihiko et al disclose a rate matcher that is performed after interleaving (See pp. 325, "Figure 1, Burst Error Reduction methods"). Therefore, it would have been obvious ... to incorporate a rate matcher that is performed after interleaving as taught by Akihiko et al into the invention of Chen et al and Frenger et al to rate match *for reducing burst errors*." (Emphasis added).

As stated above, the Main Brief filed August 11, 2006 fully identified and discussed the deficiencies of Chen et al. and Frenger et al., and this Reply Brief incorporates that discussion by reference.

Regarding the motivation alleged in this section of the Examiner's Answer, the Appellant submits that the motivation of "burst error reduction" would not have led a person skilled in the art at the time the present invention was made to incorporate the rate matching of Frenger et al. into Chen et al. This is because the main function of rate matching is to adjust the data length, not to reduce burst error. The reduction of burst error is achieved by the interleaving, and the interleaving is disclosed in Chen et al. without the need to refer to Frenger et al. Thus, the reduction of burst error is neither a motivation nor a rationale for combining the rate matching of Frenger et al. into Chen et al. It is submitted that the use of this rationale or motivation in the Examiner's Answer is unwarranted and is based on improper hindsight.

With respect to the newly applied Akihiko Watanabe et al. reference, the Appellant notes that, in this reference, the rate matching is performed for adjusting the data length to a frame length, and there are a variety of frame lengths depending on the service and data to be transmitted. Therefore, the rate-matching needs to be adapted to repeat or puncture a part of the bits of a bit sequence as well as to repeat every bit of the bit sequence. In the case that every bit is repeated, the unequalizing of the resistibility to the error does not arise due to the interleaving because there is no difference in the resistibility to the error for each bit (because there are no non-repeated bits). On the other hand, in the case that only a part of the bits is repeated or punctured, the unequalizing of the resistibility to the error arises due to the interleaver because there is a difference in the resistibility to the error between bits (because some bits are repeated and others are not; that is, the repeated bits have higher resistance). In other words, a repeated bit has high resistibility to error compared with a non-repeated bit, and a non-punctured bit has high resistibility to the error compared with a punctured bit. Thus, it could arise that bits which have low resistibility to error are gathered in a portion of the bit sequence (unequalizing of the resistibility to the error) due to the interleaving.

The present inventor recognized the problem of the unequalizing of the resistibility of the error in the case that a part of the bits is repeated or punctured as explicitly recited in independent claims 11, 19, 23 and 31.

On the other hand, in Akihiko Watanabe et al., the repetition repeats every bit as described in lines 12-14 in the left column of page 325. This is also clear from Fig. 1 which shows 64 Kbps as output from the repetition of input bits as 32 Kbps. That is, Akihiko Watanabe et al. repeat every bit and never a part of the bit sequence. Therefore, in Akihiko Watanabe et al., there is no recognition of the problem to be solved by the claimed invention, so that there is no advantage of performing the interleaving before rate matching. Actually, Watanabe's Figures show that the repetition followed by interleaving (Rep-Int method) is superior to the interleaving followed by the repetition (Int-Rep method).

Moreover, Akihiko Watanabe et al. do not disclose the rate matching defined in the Appellant's claims which comprise alternatively repeating and puncturing.

In summary, there is no motivation to combine the rate matcher of Akihiko Watanabe et al. into the systems of Chen et al. and Frenger et al. Chen et al. do not teach or suggest the adapting the various data rates. Chen et al. and Frenger et al. do not teach or suggest the increasing the resistibility to burst error. There is no motivation to replace the rate-matcher of Frenger et al. with the rate-matcher of Watanabe et al. Reducing burst error cannot be a motivation to combine these references. The rate-matcing is used for adapting the data rate not for reducing burst error. The rate-matching of Frenger et al. is performed within a coder (RCPC) or repeater concatenated with a convolutional coder. The rate-matching is performed within or immediately after error correction coding if combined. The rate-matching of Akihiko Watanabe et al. can not be adapted to various data rates. Every bits not a part of bits is repeated twice (adapted only to double the data length). Akihiko Watanabe et al. do not teach a problem of unequalizing (concentrating) of repeated bits. There is no problem of unequalizing of the repeated bits because every bits is repeated. Akihiko Watanabe et al. teach that the interleaving after repeating has good performance for the burst error compared with the interleaving before repeating. If a person skilled in the art intends to reduce the burst error, the interleaving would be performed after rate-matching as taught by Akihiko Watanabe et al.

The difference of the effect from the claimed invention would be due to whether every bits or a part of bits is repeated.

Further, as recited in dependent claims 13 and 25, the repetition and the puncturing of the part of the bits are performed at regular intervals in order to adequately equalize the resistibility to the error for the bit sequence in the case that a part of the bits is repeated or punctured. None of the applied references discloses this feature, and thus these claims provide a further basis for allowability.

For at least the above reasons and those set forth in the Main Brief, it is submitted that the new ground of rejection of claims 11-13, 19, 21-25, 31, 33, 34, 36, 38, 41 and 43 over Chen et al., Frenger et al., and Akihiko Watanabe et al. is unwarranted and should be reversed.

Thus, it is submitted that claims 23 and 31 and all claims dependent therefrom are allowable for the reasons given in the main Brief and in the discussion above.

3. Rejection of Claims 20, 32, 35, 37, 39, 40, 42 and 44 under 35 USC 103(a)

The Examiner's Answer acknowledges that the teachings of Chen et al., Frenger et al., and Akihiko Watanabe et al. fail to disclose, on the reception side, employing a second rate matcher that comprises a second repeater and a second puncturer to alternately select between repeating and puncturing and employing de-interleaving of data including bits provided by the second rate matcher. The paragraph bridging pages 7 and 8 of the Examiner's Answer state that "However, the admitted prior art, in the receiver side, teaches that a signal received by a reception antenna is subjected to predetermined radio processing and demodulation processing and so forth before inverse rearrangement against the interleaving is performed in the de-interleaving. In this rearranged data, the number of bits which are increased or decreased in the transmitter side once are decreased or increased in the puncturing or repeating section." The Examiner's answer states in the first full paragraph of page 8 that "Therefore, it would have been obvious ... to incorporate the steps of employing a second rate matcher that comprises a second repeater and a second puncturer to alternatively select between second repeater and second puncturer, and performing de-interleaving of data as taught by the AAPA into the invention of Chen et al, Frenger et al and Akihiko et al in order for adjusting coded data to frame length. The admitted prior art, in the receiver side, does not disclose that rate matcher follows by de-interleaving. However, since, in the transmitter side, the order of interleaving before or after rate matcher is obvious ... to reduce burst error. Therefore, it would have been obvious ... in the receiver side, to reconstruct the interleaving data by de-interleaving it before or after a second rate matching by repetition or puncture, according to the order of the transmitter side, for adjusting the number of bits in the data block to reverse the action of the coding device. As per claims 32, 35, 37, 39-40, 42 and 44, these claims are rejected under similar rationale as set forth in claim 20."

The rejection of the reception claims (claims 20, 32, 35, 37, 39, 40, 42 and 44) asserts, in effect, that it would have been obvious to incorporate the steps of employing a second rate matcher that punctures bits repeated by the rate matcher of the transmission method and performing deinterleaving of data including bits provided by the second rate matcher, as taught by the admitted prior art, into the invention of Chen et al and Frenger et al in order for adjusting coded data to frame length. The alleged motivation for the combination is to adjust coded data to frame length. However, this statement provides no reason for selecting a timing for the rate matching, i.e., either before or after deinterleaving. The statement merely assumes that the rate matching occurs before the deinterleaving step. Thus, the alleged motivation ignores a critical aspect of the invention relating to the order of the deinterleaving and rate matching steps. The deficiencies of Akihiko Watanabe et al. have been addressed above. It is further noted that Akihiko Watanabe et al. do not address deinterleaving, and, as discussed above, in Akihiko Watanabe et al., there is no recognition of the problem to be solved by the claimed invention, so that there is no advantage of performing deinterleaving and rate matching in any particular order.

For the reason given above and in the main Brief, it is submitted that claims 20, 32, 35, 37, 39, 40, 42 and 44 are allowable over the individual or combined teachings of the applied art.

4. Section 10: Response to Argument

Regarding this Section of the Examiner's Answer, first of all, it is unclear what is meant by the first and second paragraphs of this section. If the Examiner's Answer is proposing that Appellant's argument is unsupported that Frenger et al. does not disclose a single device employing both repeating and puncturing, the Examiner's Answer is based on an explicit misunderstanding of Frenger et al. Frenger et al. is primarily directed to use of rate-compatible punctured convolutional codes (RCPC-codes), and discloses a convolutional code concatenated with a repetition code for purposes of showing the superiority of the RCPC-codes. This is clear from the description in the Abstract and Section V ("COMPARISON TO REPETITION ENCODING") of Frenger et al. Therefore, in the view of one of ordinary skill in the art, it does make sense to take a position that Frenger et al. does not contemplate that both repeating and puncturing are employed with a single device and does not contemplate to alternatively select between repeating and puncturing as a positive (mandatory) operation in the rate matching.

Regarding the third paragraph of Section 10, it is noted that the second "sentence" of this paragraph is incomplete and unable to be understood by the Appellant. Further, the basis for the allegations in the third and fourth sentences is unclear. The rejections should not be sustained based upon such unclear statements.

In the paragraph bridging pages 9 and 10, The Examiner's Answer proposes that "Therefore, one of ordinary skill ... would have known that Frenger uses only one convolutional encoder with embedded rate matcher that comprises a puncture and a repetition, wherein the rate matcher alternatively selects between repetition and puncture. (See pg. 354, section II: RCPC-Codes for Rate Matching)." However, as mentioned above, such subject matter has not been known prior to the present invention. Section II of Frenger et al. cited by the Examiner's Answer describes that "An alternative way to perform rate matching is to have a higher rate convolutional encoder concatenated with a repetition encoder that simply repeats some of the bits before transmission." It should be noted that the higher rate convolutional encoder is different from the RCPC-Codes in Frenger et al., so that it is unclear how the higher rate convolutional encoder is merged with the RCPC-Codes and how the repetition concatenated with higher rate convolutional encoder is operable in the RCPC-Codes.

In the third full paragraph of page 10, the Examiner's Answer states that "Examiner disagrees with applicant because nothing spectacular about placing interleave before or after since it serves a distinct purpose from rate matching is for burst error." First of all, this sentence is unclear as to the intended meaning. Secondly, if the Examiner's Answer is stating that the purpose of rate matching is different from that of the interleaving which is used for reducing bust error, it weakens the Examiner's argument concerning combining Frenger et al. with Chen et al. That is, as mentioned above, the reduction of burst error cannot be a motivation for combining the rate matching taught by Frenger et al. with Chen et al.

Regarding the Examiner's argument described in the fifth full paragraph of page 10, the Appellant notes that the only motivation to conceive the claimed invention is to find out the unequalizing occurred due to the interleaving performed after the rate matching and to strengthen the resistibility to the burst error. However, Akihiko Watanabe et al., as well as the previously cited references, fail to provide such a motivation.

At page 11, the Examiner's Answer again notes that Akihiko Watanabe et al. disclose rate matching performed after interleaving, and that it would have been obvious to incorporate this teaching into the invention of Chen et al. and Frenger et al. to rate match for reducing burst errors. The Examiner's Answer argues that the motivation is provided by Akihiko Watanabe et al. at page 325, Fig. 1, which shows three Burst Error Reduction Methods. However, as noted above, the present inventor recognized the problem of the unequalizing of the resistibility of the error in the case that a part of the bits is repeated or punctured as explicitly recited in independent claims 11, 19, 23 and 31. On the other hand, in Akihiko Watanabe et al., the repetition repeats every bit as described in lines 12-14 in the left column of page 325. This is also clear from Fig. 1 which shows 64 Kbps as output from the repetition of input bits as 32 Kbps. That is, Akihiko Watanabe et al. repeat every bit and never a part of the bit sequence. Therefore, in Akihiko Watanabe et al., there is no recognition of the problem to be solved by the claimed invention, so that there is no advantage of performing the interleaving before rate matching. Actually, Watanabe's Figures show that the repetition followed by interleaving (Rep-Int method) is superior to the interleaving followed by the repetition (Int-Rep method).

Moreover, Akihiko Watanabe et al. do not disclose the rate matching defined in the Appellant's claims which comprise alternatively repeating and puncturing.

Further, as recited in dependent claims 13 and 25, the repetition and the puncturing of the part of the bits are performed at regular intervals in order to adequately equalize the resistibility to the error for the bit sequence in the case that a part of the bits is repeated or punctured. None of the applied references discloses this feature, and thus these claims provide a further basis for allowability.

In view of the above arguments and those made in the main Brief, it is submitted that the rejections of all pending claims are unwarranted, and it is requested that this honorable Board reverse the same.

Respectfully submitted,

James E. Ledbetter
Registration No. 28,732

Date: January 16, 2007
JEL/att
ATTORNEY DOCKET NO. L9289.00121
STEVENS, DAVIS, MILLER & MOSHER, L.L.P.
1615 L Street, NW
P.O. Box 34387
Washington, DC 20043-4387
Telephone: (202) 408-5100
Facsimile: (202) 408-5200

Evidence Appendix

Copy of Hagenauer article ("Rate-Compatible Punctured Convolutional Codes (RCPC Codes) and their Applications")

Rate-Compatible Punctured Convolutional Codes (RCPC Codes) and their Applications

JOACHIM HAGENAUER, SENIOR MEMBER, IEEE

Abstract—**The concept of punctured convolutional codes is extended by puncturing a low rate $1/N$ code periodically with period P to obtain a family of codes with rate $P/(P + l)$ where l can be varied between 1 and $(N - 1)P$. A rate-compatibility restriction on the puncturing tables ensures that all code bits of high rate codes are used by the lower rate codes. This allows transmission of incremental redundancy in ARQ/FEC schemes and continuous rate variation to change from low to high error protection within a data frame. Families of RCPC codes with rates between 8/9 and 1/4 are given for memories M from 3 to 6 (8 to 64 trellis states) together with the relevant distance spectra. These codes are almost as good as the best known general convolutional codes of the respective rates. It is shown that the same Viterbi decoder can be used for all RCPC codes of the same M by controlling the metric memory access through the puncturing rule, the soft decision and the channel state information (CSI). The performance of RCPC codes on Gaussian and fading channels is shown under various quantization and CSI conditions. The application of RCPC codes to hybrid ARQ/FEC schemes is discussed for Gaussian and Rayleigh fading channels using CSI to optimize throughput. If a source coder supplies a source significance information (SSI) indicating the relative importance of bits, a flexible unequal error protection scheme can be designed with RCPC codes which minimizes the redundancy overhead.**

I. Introduction

The design of an error correction coding system usually consists of selecting a fixed code with a certain rate and correction capability matched to the protection requirement of all the data to be transmitted and adapted to the average or worst channel conditions to be expected [1]. In many cases, however, one would like to be more flexible because the data to be transmitted have different error protection needs and the channel is time varying or has insufficiently known parameters. Consequently, flexible channel encoding and an adaptive decoder are required. As Fig. 1 shows, the information to be transmitted might carry source significance information (SSI) indicating different protection requirements. Examples include control and signaling bits in a data stream, most significant bits in a PCM word, or side information like quantizer slopes, or filter coefficients in more sophisticated voice encoding schemes [2]. On the other hand, the channel characteristics or the channel state might vary considerably, as encountered in mobile or multipath radio transmission, in a jamming environment, during rain fading, or in HF transmission. This is indicated in Fig. 1 by the channel state information CSI. In rare cases the instantaneous CSI is available at the encoder where code adaptation could take place. Mostly the receiver only can use a CSI-like fading depth, noise level variation, short-term signal loss, or jammer activity. The CSI can significantly improve decoder performance together with soft decisions at the receiver. Whenever a return channel is available, the CSI can be indirectly relayed to the transmitter by asking for a retransmission as employed in automatic repeat request (ARQ) systems. Such ARQ systems can also be combined with forward error correction (FEC) to yield type I or type II ARQ/FEC hybrid schemes [3], [4]. The latter schemes nicely match the average channel rate or the throughput with the channel conditions and some of them use different FEC codes for repeated transmission attempts. The scenarios shown in Fig. 1 require variable codes adapted to the source and channel needs. We wish to change the code rate, i.e., the number of check bits, and hence the correction power of the code during transmission of an information frame according to source and channel needs. For practical purposes, we would like to have not just switching between a set of encoders and decoders, but one encoder and one decoder which can be modified without changing their basic structure. This can be achieved by not transmitting certain code bits, namely, by puncturing the code. Mandelbaum [5] was the first to propose punctured codes for transmitting redundancy in incremental steps by using Reed–Solomon codes. In order to accommodate soft decisions and CSI at the receiver, a maximum likelihood decoder is required. This motivates the use of convolutional codes and the Viterbi algorithm for decoding.

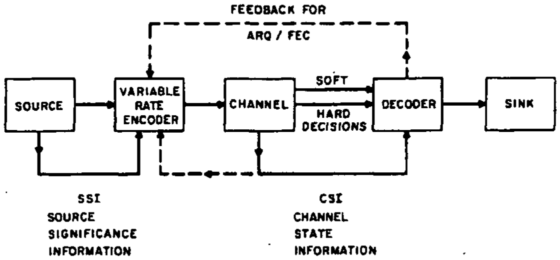


Fig. 1. Coded transmission scheme with source significance information (SSI) and channel state information (CSI).

Punctured convolutional codes were first introduced by Cain, Clark, and Geist [6] mainly for the purpose of obtaining simpler Viterbi decoding for rate K/N codes with two branches arriving at each node instead of 2^K branches. They obtained codes of rate 2/3 and 3/4 by puncturing rate 1/2 codes. These punctured codes were almost as good as the best known codes. Some of the good codes used the same basic rate 1/2 generators. Later, Yasuda *et al.* [7], [8] found a family of $(N - 1)/N$ codes by puncturing 1/2 codes for N up to 14, and built selectable rate encoders and Viterbi decoders using soft decisions.

In this paper, the concept of punctured convolutional codes is modified for the generation of a family of codes by adding a rate-compatibility restriction to the puncturing rule. The restriction implies that all the code bits of a high rate punctured

Paper approved by the Editor for Coding Theory and Application of the IEEE Communications Society. Manuscript received December 22, 1986; revised June 5, 1987. This paper was presented in part at the 1986 International Symposium on Information Theory, Ann Arbor, MI, October 1986, and ICC'87, Seattle, WA, June 1987.

The author is with the German Aerospace Research Establishment (DFVLR), D-8031 Oberpfaffenhofen, West Germany.

IEEE Log Number 8718991.

code are used by the lower rate codes; or in other words, the high rate codes are embedded into the lower rate codes of the family. If the higher rate codes are not sufficiently powerful to decode channel errors, only supplemental bits which were previously punctured have to be transmitted in order to upgrade the code. Furthermore, since codes are compatible, rate variation within a data frame is possible to achieve unequal error protection. In Section II, the necessary definitions and the performance criterion for the class of rate compatible punctured convolutional codes (RCPC codes) are given. Section III reports the results of a computer search for families of RCPC codes of rate $P/(P + l)$, with rates between 1/4 and 8/9. The analytically calculated and simulated performance of some codes on Gaussian, Rayleigh, and Rice fading channels is shown in Section IV. Modified type II ARQ/FEC schemes with RCPC schemes shown in Section V allow a minimum number of transmitted bits as well as soft decision and CSI adaptive decoding and compare favorable to previous known schemes. Unequal error protection with RCPC codes is discussed in Section VI.

II. Definition and Performance Criterion of RCPC Codes

In order to explain rate-compatible punctured codes, we start with the example of Fig. 2 where a rate $R = 1/2$ convolutional code with memory $M = 2$ is punctured periodically with period $P = 4$. We describe the binary information symbols and the binary code symbols x_{1j} and x_{2j} by values ± 1. A zero in the puncturing table means that the code symbol is not to be transmitted. In the upper example of Fig. 2, the fourth bit of the upper branch and the second and third bit of the lower branch are not transmitted. The puncturing table can be viewed as a modulo P rule for multiplexing the two streams of code bits. Instead of transmitting $2 \cdot P = 2 \cdot 4$ only $P + l = P + 1 = 5$ bits are transmitted per $P = 4$ information bits. Therefore, we have generated a code with rate $R = 4/5$ by using only $P + l$ "1" 's instead of $2P$ "1" 's in the puncturing table, which can be described by the $N \times P$ matrix

$$a(1) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}.$$

Suppose the code rate with 4/5 and with the puncturing table $a(1)$ is not powerful enough to correct the channel errors. A more redundant and therefore more powerful code with lower rates 4/6, 4/7, or 4/8 would be necessary. Instead of transmitting all the code bits of a completely different low rate code, the lower rate code should utilize the bits already transmitted. Then only additional incremental redundancy bits have to be transmitted. Additional "1" 's in the puncturing tables of the lower rate codes can therefore be placed only where zeros appeared in the puncturing matrix of the previous higher rate code, for example,

$$a(2) = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}, \quad a(3) = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{pmatrix},$$

$$a(4) = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

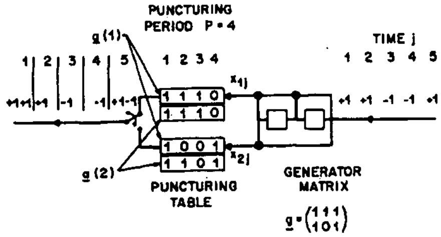


Fig. 2. Example of a punctured convolutional code with two rate compatible puncturing tables.

Thus, for $l = 1$ to 4, we get a compatible family of codes derived from the mother code 1/2 with rates 4/5, 4/6, 4/7, 4/8 which use only incremental redundancy. The question, of course, is whether all these codes in the family are good and whether noncatastrophic codes exist. This will be discussed in Section III. On the receiving side the decoder using the Viterbi algorithm (VA) has to know the current puncturing rule a. The VA receives real values y_{ij} when full soft decision is used. Alternatively, the decoder can use quantized values of y_{ij}. The VA finds the path with the maximum likelihood metric for statistically independent y_{ij} by calculating

$$\max_m \sum_{j=1}^{J} \lambda_j \tag{1}$$

where the metric increment is

$$\lambda_j = \sum_{i=1}^{N} a_{ij} x_{ij}^{(m)} y_{ij} \tag{2}$$

and $a_{ij+P} = a_{ij}$ due to periodic puncturing. $a_{ij} = 0$ means that this $x_{ij}^{(m)}$ has not been transmitted and y_{ij} is not available. Actually, it is not necessary to insert any dummy data at the receiver as suggested in [7]. The Viterbi algorithm operates on the trellis of the $1/N$ mother code with two branches leaving at each state. Equation (2) simply means that for $a_{ij} = 1$ the metric ROM is accessed using the address $x_{ij}^{(m)} y_{ij}$ to add a metric increment and for $a_{ij} = 0$ it is not accessed.

For a fading channel, the transmitted signal is multiplied by a fading factor a_F. If interleaving is assumed, a_F is an independent positive real random variable with density $p(a_F)$, i.e., Rayleigh or Rice. If this factor can be estimated at the receiver and is used as a CSI the ML-metric for coherent transmission is [9]

$$\lambda_j = \sum_{i=1}^{N} a_{ij} a_{ijF} x_{ij}^{(m)} y_{ij}. \tag{3}$$

As expected, a deep fade ($a_{ijF} \rightarrow 0$) has the same effect as puncturing ($a_{ij} = 0$).

A. General Definition of RCPC Codes

A family of RCPC codes is described by the mother code of rate $R = 1/N$ and memory M having the generator tap matrix

$$g = \overset{\leftarrow M+1 \rightarrow}{N \updownarrow \left(g_{ik} \right)} \tag{4}$$

with the tap connections $g_{ik} \in (0, 1)$ where a 1 represents a connection from the kth shift register stage to the ith output. Together with N, the puncturing period P determines the range of code rates

$$R = \frac{P}{P+l} \qquad l = 1, \cdots (N-1)P \tag{5}$$

between $P/(P + 1)$ and $1/N$. The RCPC codes are punctured codes of the mother code with puncturing matrices

$$a(l) = \overset{\leftarrow P \rightarrow}{N \updownarrow \left(a_{ij}(l) \right)} \tag{6}$$

with $a_{ij}(l) \in (0, 1)$ where 0 implies puncturing.

The rate-compatibility restriction implies the following rule:

$$\text{if } a_{ij}(l_0)=1 \text{ then } a_{ij}(l)=1 \text{ for all } l \geqq l_0 \geqq 1 \quad (7a)$$

or equivalently

$$\text{if } a_{ij}(l_0)=0 \text{ then } a_{ij}(l)=0 \text{ for all } l \leqq l_0 \leqq (N-1)P-1. \quad (7b)$$

B. Criterion of Goodness for RCPC Codes

With Viterbi decoding the usual optimality criterion is a large free distance d_{free}, a small number of paths a_d, and a small information error weight c_d on all paths with $d \geqq d_{\text{free}}$. More specifically, one has Viterbi's upper bound an error event probability [13]

$$P_E \leqq \frac{1}{P} \sum_{d=d_{\text{free}}}^{\infty} a_d P_d \quad (8)$$

and for the bit error probability

$$P_b \leqq \frac{1}{P} \sum_{d=d_{\text{free}}}^{\infty} c_d P_d. \quad (9)$$

P_d is the probability that the wrong path at distance d is selected. The so-called distance spectra $\{a_d\}$ and $\{c_d\}$, which should be as small as possible, depend only on the code, namely, on N, M, P, g, and $a(l)$. Due to the time-varying nature of the RCPC codes, P different starting points for diverging paths have to be considered to obtain the total numbers a_d and c_d. Since RCPC codes are a subclass of punctured codes, which in turn is a subclass of the general K/N codes, it has still to be shown that the RCPC restriction is not too severe to prevent "good" codes from being located in this subclass. A degree of freedom can be gained by increasing the period P. RCPC codes constitute a specific class of time-varying codes with a fixed generator but periodically time-varying puncturing. We know that some time-varying codes are better than fixed codes [10] and some random coding theorems hold only for time-varying codes. Therefore, some improvement may be possible by using a higher period P. Note that P has to be sufficiently large if codes of specific rates are to occur in the same RCPC family. The complexity of the VA decoder of an RCPC code in terms of add, compare and select operations per decoded bit is $2 \cdot 2^M$, whereas a general K/N code requires $(2^K 2^M)/K$ such operations where M is the total encoder memory. At the RCPC decoder, an $N \cdot P$ ambiguity has to be resolved in the incoming data stream and the decision depths of punctured codes are generally longer [8]. The last disadvantage is not severe, when as in most applications, data frames are transmitted with sync words and proper termination of short frames.

III. Results of the Search for Good RCPC Codes

No constructive method is known for determining the generator matrix g and the puncturing matrices $a(l)$ for a RCPC code family. Therefore a computer search has been performed under some restrictions. For distances $d \geqq d_{\text{free}}$, a_d, c_d, and the path length distribution has been determined by a stack algorithm similar to Cedervall's algorithm [11].

The basic steps to find a family of good RCPC codes are:

Step 1: Select mother code with rate $1/N$, memory M, and generator g. Select puncturing period P. For $l = l_{\max} = (N-1)P$ set $a(l) = (1)$.

For $l = (N-1)P$ to 2 perform steps 2–4.

Step 2: Amongst the elements $a_{ij}(l)$ which are "1" select one and set it to "0" [compatibility rule (7b)]. Observe that cyclic column permutations give the same code performance.

Step 3: Determine d_{free} and c_d for $d \geqq d_{\text{free}}$.

Step 4: Repeat steps 2 and 3 to find $a(l-1)$ with maximal d_{free} and minimal c_d for $d = d_{\text{free}}$, $d_{\text{free}} + 1$. If only catastrophic codes can be found, use next best code at level l.

The results of the computer search for the $M = 4$ code family are given in some detail in Table I, whereas for the other codes of memory $M = 3, 5, 6$, only $a(l)$ and the c_d values are given in Table II. The codes have a period $P = 8$ with a subperiod of $P = 4$ and are punctured mostly in steps $\Delta l = 2$ in such a way as to arrive at the best known code of rate 1/2. This implies that the mother codes of rate 1/3 or 1/4 are not necessarily the best known code of this rate. This is the case for $M = 4$ and 6 where one path exists at a distance which is one less than the best known free distance [3].

Table III shows a comparison of the most important distance parameters of RCPC codes, punctured codes, and general K/N codes for rates where results are available. This and other comparisons show that RCPC codes are almost as good as the other codes. By increasing the period P one can even obtain slightly better codes as shown for $M = 3$ and rate 2/3 code. $P = 4$ gives a greater variety in selecting the puncturing positions. In this case, c_d/P for $d = d_{\text{free}}, \cdots, d_{\text{free}} + 5$ is less than the corresponding numbers for the code with $P = 2$ and therefore a better BER performance can be expected over the whole SNR range. This result is a further confirmation that time-varying codes are better than fixed codes and that time-varying codes become better with increasing time period. In [10], an example is given where g is time varying. Here, we fix g and puncture in a time-varying manner.

One disadvantage of punctured codes is that error events at high rates and higher distances can be quite long. This was previously observed in [7]. To give an example, we have plotted in Fig. 3 the path length distribution of an $R = 4/5$ code with $M = 4$, $P = 4$. Paths at $d_{\text{free}} + 3 = 6$ can be up to 50 information bits long. For high rate codes, this requires a longer path memory.

IV. Performance of RCPC Codes on Gaussian and Fading Channels

The performance of RCPC codes on a Gaussian channel with soft decision and on interleaved Rayleigh and Rician channels using different degrees of channel state information has been evaluated using (9) and the distance spectrum $\{c_d\}$ in Tables I and II. For the Gaussian case simulation results are also given. Fig. 4 shows the transmission scheme over a nonfrequency-selective fading channel with multiplicative distortion a_F, with $E\{a_F^2\} = 1$ and

$$p_{a_F}(a_F) = 2a_F(1 + C/M) \exp\left[-(a_F^2(1 + C/M) + C/M)\right] \cdot I_0(2a_F\sqrt{C/M(1 + C/M)}). \quad (10)$$

For a Ricean channel, C/M is the ratio of the direct to diffusely reflected signal energy. With $C/M = 0$ we have Rayleigh fading, with $C/M \to \infty$ we obtain the Gaussian channel. In the fading case we assume perfect interleaving, which means that the CSI values a_F and the received code values y are statistically independent with density function (10). Since we consider a varying code rate, we describe the channel SNR by E_S/N_0 rather than by $E_b/N_0 = E_S/RN_0$. A reduced rate consequently means reduced throughput. The Viterbi algorithm for decoding the RCPC code uses the metric (3). In the following, we give the values for $\hat{a}_F = a_{ijF}$ and $y = y_{ij}$ which the decoder uses in different cases, as well as the resulting P_d to be used in the evaluation of the BER in (9).

a) Gaussian channel [13]

$$\hat{a}_F = 1 \qquad y \text{ analog or finely quantized}$$

$$P_d = \frac{1}{2}\, erfc\sqrt{dE_S/N_0}.$$

TABLE I
PUNCTURING TABLES $a(l)$, c_d, AND a_d VALUES FOR RCPC CODES WITH MEMORY $M = 4$. PERIOD $P = 8$ AND RATES $R = P/(P + l) = 8/(8 + l)$, $l = 1, 2, 4, 6, \cdots, 24$

RCPC CODE: M = 4 P = 8

c_d

d	8/9 8/9 1111 0111 1000 1000 0000 0000	8/10 4/5 1111 1111 1000 1000 0000 0000	8/12 2/3 1111 1111 1010 1010 0000 0000	8/14 4/7 1111 1111 1110 1110 0000 0000	8/16 1/2 1111 1111 1111 1111 0000 0000	8/18 4/9 1111 1111 1111 1111 1000 1000	8/20 4/10 1111 1111 1111 1111 1100 1100	8/22 4/11 1111 1111 1111 1111 1110 1110	8/24 1/3 1111 1111 1111 1111 1111 1111	8/26 4/13 1111 1111 1111 1111 1111 1111 1000 1000	8/28 2/7 1111 1111 1111 1111 1111 1111 1010 1010	8/30 4/15 1111 1111 1111 1111 1111 1111 1110 1110	8/32 1/4 1111 1111 1111 1111 1111 1111 1111 1111
2	1												
3	242	42											
4	4199	274	4										
5	63521	2688	0	2									CODE 10011 11101 10111 11011
6	885318	21692	496	62									
7	11678199	154684	0	144	32	2							
8		1103894	10884	350	96	36	2						
9			0	2006	160	60	34	10					
10				5394	576	82	28	8					
11					1800	354	66	38	8	2			
12					4000	856	226	72	48	4			
13							354	114	72	56	20	2	
14								228	48	40	20	16	
15									104	38	36	38	32
16									256	104	24	0	16
17											56	18	0
18											184	74	32
19													48
20													96

a_d

d	8/9	8/10	8/12	8/14	8/16	8/18	8/20	8/22	8/24	8/26	8/28	8/30	8/32
2	1												
3	30	8											
4	327	40	4										
5	3493	274	0	2									
6	37729	1686	108	18									
7	406015	9842	0	32	16	2							
8		59406	1380	74	24	14	2						
9			0	308	32	18	14	4					
10				696	80	22	10	6					
11					298	76	20	14	8	2			
12					544	164	54	22	18	4			
13							78	30	24	20	12	2	
14								52	16	14	12	8	
15									24	12	12	18	18
16									64	30	8	0	8
17											18	14	0
18											52	24	8
19													16
20													32

b) Rayleigh and Rice fading with hard decisions on y (YH) and no CSI, a not used (AN): YHAN

$$\hat{a}_F = 1, \quad y = \pm 1.$$

$$P_d = \begin{cases} \sum_{(d+1)/2}^{d} \binom{d}{e} P_0^e (1-P_0)^{d-e} & d \text{ odd} \\ P_{d-1} & d \text{ even} \end{cases} \qquad (11)$$

with the channel bit error rate

$$P_0 = \int_0^\infty \frac{1}{2}\, erfc\sqrt{a_F^2 \cdot E_S/N_0}\, p_{a_F}(a_F)\, da_F. \qquad (12)$$

c) Rayleigh fading with soft decisions on y (YS) and full CSI (AS) with perfect estimation of the analogy amplitude a_F: YSAS

$$\hat{a}_F = a_F \qquad y \text{ analog or finely quantized}$$

P_d is equivalent to the error probability of a dth order maximum ratio diversity system and can be upper bounded by [9]

$$P_d \leq \frac{1}{2}\left(\frac{1}{1+E_S/N_0}\right)^d. \qquad (13)$$

d) Rayleigh fading with hard decision (YH) and hard CSI

TABLE II

PUNCTURING TABLES $a(l)$ AND c_d VALUES FOR RCPC CODES WITH MEMORY $M = 3, 5, 6$.
PERIOD $P = 8$ AND RATES $R = P/(P + l) = 8/(8 + l)$, $l = 1, 2, 4, 6, \cdots, 14, 16$

RCPC CODE: M = 3

d	8/9	8/10	8/12	8/14	8/16	8/18	8/20	8/22	8/24	
	8/9	4/5	4/6	4/7	1/2	4/9	2/5	4/11	1/3	
	1110 1110 1001 0001 0000 0000	1110 1110 1001 1001 0000 0000	1110 1110 1101 1101 0000 0000	1111 1111 1101 1101 0000 0000	1111 1111 1111 1111 0000 0000	1111 1111 1111 1111 1000 1000	1111 1111 1111 1111 1010 1010	1111 1111 1111 1111 1110 1110	1111 1111 1111 1111 1111 1111	
2	21									CODE
3	509	62								1101
4	7759	528	32	4						1111
5	99172	3894	156	0						1011
6	1168056	26580	736	160	16	4				
7	13088533	171784	2990	0	56	22	8			
8		1070980	12030	1806	144	34	16	10		
9			46688	0	392	118	24	20		
10					1040	286	92	22	48	
11					2664	580	200	58	0	
12							336	140	48	
13								284	0	
14									464	
15									0	
16										
17										
18										
19										
20										

(a)

RCPC CODE: M = 5

d	8/9	8/10	8/12	8/14	8/16	8/18	8/20	8/22	8/24	
	8/9	4/5	2/3	4/7	1/2	4/9	2/5	4/11	1/3	
	1111 1111 1000 0000 0000 0000	1111 1111 1000 1000 0000 0000	1111 1111 1010 1010 0000 0000	1111 1111 1110 1110 0000 0000	1111 1111 1111 1111 0000 0000	1111 1111 1111 1111 1000 1000	1111 1111 1111 1111 1010 1010	1111 1111 1111 1111 1110 1110	1111 1111 1111 1111 1111 1111	
2										CODE
3	231									111101
4	4229	80								101011
5	57612	762								100111
6	850396	6502	384	10						
7	11342400	54246	0	118						
8	151893233	426732	7616	316	16	4				
9		3239744	0	720	288	12				
10			143744	3048	256	100	8			
11			0	10438	496	322	100	2		
12					2656	234	144	32		
13					5608	796	292	134	8	
14							368	148	64	
15							828	264	208	
16								384	160	
17									152	
18									496	
19										
20										

(b)

TABLE II (*Continued*)

RCPC CODE: M = 6

d ↓	8/9 8/9 1111 0111 1000 1000 0000 0000	8/10 4/5 1111 1111 1000 1000 0000 0000	8/12 2/3 1111 1111 1010 1010 0000 0000	8/14 4/7 1111 1111 1110 1110 0000 0000	8/16 1/2 1111 1111 1111 1111 0000 0000	8/18 4/9 1111 1111 1111 1111 1000 1000	8/20 4/10 1111 1111 1111 1111 1100 1100	8/22 4/11 1111 1111 1111 1111 1110 1110	8/24 1/3 1111 1111 1111 1111 1111 1111	
2										CODE
3	24									1011011
4	740	24								1111001
5	13321	376								1100101
6	217761	3464	12							
7	3315491	30512	280	12						
8	48278177	242734	1140	74						
9		1890790	5104	388						
10			24640	1162	288	14				
11			108512	3542	0	82	8			
12				11666	1688	182	44	2		
13					0	320	80	16		
14					11232	810	138	70	8	
15					0	1882	444	90	0	
16							750	172	160	
17								474	0	
18									424	
19									0	
20										

(c)

TABLE III
COMPARISON OF RCPC CODES WITH PUNCTURED AND GENERAL CONVOLUTIONAL CODES

Rate	Memory	RCPC-Codes			Punctured Codes [8]			General K/N Codes
		d_{free}	$c_{d_{free}}/P$	P	d_{free}	$c_{d_{free}}/P$	P	d_{free} 1)
2/3	M = 3	4	5	2	4	$\underline{5}$	2	4 [3]
		4	4	4				
	M = 6	6	1.5	2	6	1.5	2	7 [3]
4/5	M = 3	3	$\underline{7.7}$	4	3	3.5	4	2 [12]
	M = 6	4	3	4	4	3	4	-
8/9	M = 3	2	$\underline{2.6}$	8	2	0.8	8	-
	M = 6	3	$\underline{3}$	8	3	1.6	8	-
4/7	M = 3	4	0.5	4	-	-	-	4 [12]
	M = 5	6	1.25	4	-	-	-	6 [12]

[a] With memory to give an equal number of total branches per information bit in the VA.

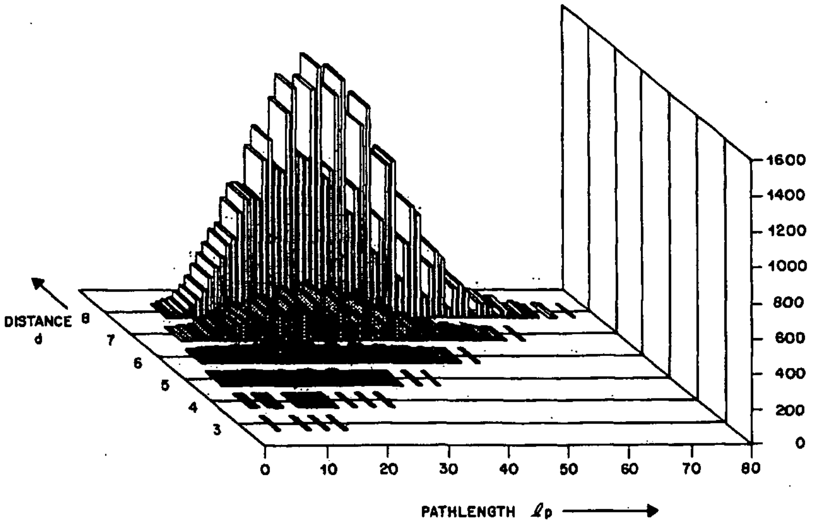


Fig. 3. Number of paths with length l_p at distance d for a RCPC code of rate $R = 4/5$, $P = 4$, $M = 4$.

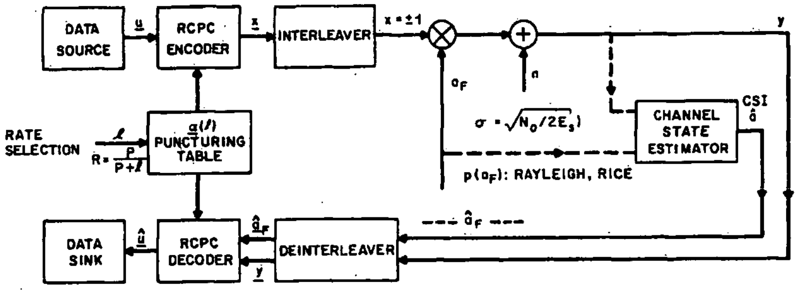


Fig. 4. Coded data transmission using RCPC codes with interleaving and channel state information (CSI) on fading channels.

(AH): YHAH

$$\hat{a}_F = \begin{cases} \log \dfrac{1-P_B}{P_B} & \text{for } a_F < a_T \\ \log \dfrac{1-P_G}{P_G} & \text{for } a_F \geqq a_T \end{cases} \tag{14}$$

where a_T is an optimized threshold. By partial integration in (12), we obtain P_G as the channel BER above the threshold a_T and P_B as the BER below a_T [9]. P_d can be calculated in closed form. The result is given in [9] and [14].

Fig. 5 shows the simulated RCPC performance for the Gaussian channel. Simulation and an analytical upper bound at a BER of 10^{-5} differ only by approximately 0.2 dB. The performance curves for the Rayleigh channel with the scheme described in Fig. 4 and in Section IV show that RCPC codes with rates between 4/5 and 1/3 cover a broad range of SNR and BER. At a BER of 10^{-5} a 1-bit CSI gains 7.9 dB at rate 2/3 and 2.6 dB at rate 1/3 relative to hard decisions without CSI. The Rice channel with $C/M = 7$ dB is typical for mobile satellite communications to ships with small antennas, to cars in open areas, and to airplanes with hemispherical antennas [14]. As Fig. 7 shows, the RCPC codes using basically the *same* encoder and the *same* Viterbi decoder with only 16 states and two branches per state can cover a SNR between 4 and 17 dB to achieve a BER of 10^{-5}. This means that the same link can serve users with a gain variation of 13 dB by adaptively changing the rate between 4/5 and 1/3 at the expense of throughput.

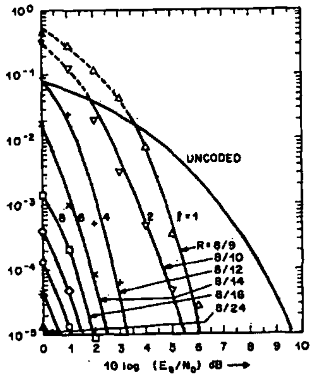


Fig. 5. BER performance of RCPC codes on Gaussian channels, $N = 3$, $P = 8$, $l = 1, 2, 4, \cdots, 16$, rate $= 8/(8 + l)$, $M = 4$: simulation with soft decision.

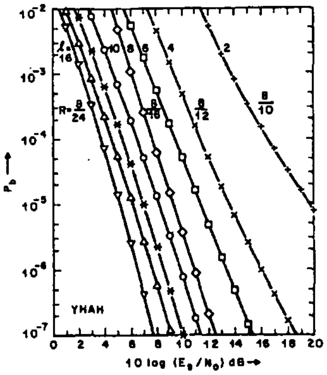


Fig. 6. BER performance of RCPC codes on an interleaved Rayleigh channel, $N = 3$, $P = 8$, $l = 2, 4, \cdots, 16$, rate $R = 8/(8 + l)$, $M = 4$; analytic upper bound YHAH: hard decision, 1 bit CSI.

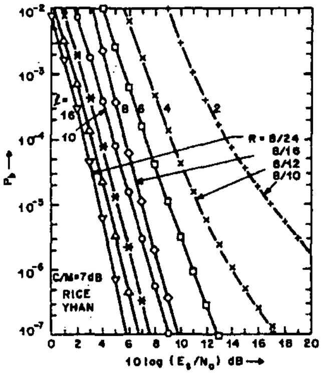


Fig. 7. BER performance of RCPC codes on an interleaved Rice channel $C/M = 7$ dB, YHAN: hard decision, no CSI, $N = 3$, $P = 8$, $l = 2, 4, \cdots, 16$, rate $R = 8/(8 + l)$, $M = 4$, analytic upper bound.

V. The Application of RCPC Codes to Hybrid ARQ/FEC Transmission

For data transmission schemes where a return channel is available, automatic repeat request (ARQ) schemes guarantee high reliability at high and moderate channel qualities. The throughput performance degrades if the channel is bad or time varying. A combination with forward error correction (FEC) to form a hybrid system combines the advantages of both methods. A broad variety of hybrid systems has been suggested, a good overview of the literature is given in [3] and [4]. Some more recent proposals can be found in [15]–[19]. A so-called "type II hybrid system" [4] uses a rate 1/2 invertible code and alternately sends coded sequences ("information" and "parity" sequences) which are either detected to be correct or combined for FEC decoding. Block and convolutional FEC codes have been used and extensions to rate 2/3 and 3/4 have been reported [4]. Typically, the "throughput-versus-channel quality"-curves exhibit a saddle at the FEC code rate and some of the schemes require different decoders at different stages of the detection and decoding algorithm. A more natural scheme than alternate code and parity transmission seems to be successive parity transmission to build up a code which is finally powerful enough to decode the message. Consequently, none of the already transmitted code bits are thrown away but are used to improve FEC decoding, preferably with ML decoding. As in most other hybrid schemes an outer high rate block-code detects decoding errors. This method goes back to Mandelbaum [5] and includes variations like code combining [15], [16] and memory ARQ [17].

We propose to use rate compatible punctured convolutional codes (RCPC Codes) with Viterbi algorithm (VA) decoding for successive parity transmission. The principle of the proposed ARQ/FEC scheme is not to repeat information or parity bits if the transmission is unsuccessful as in previous type II ARQ/FEC schemes, but to transmit additional code bits of a lower rate RCPC code until the code is powerful enough to enable decoding. This includes several decoding attempts on the receive side, which seem especially feasible for applications in mobile or low rate systems where the data rate is in the kbit/s range and the VA decoder would run on a chip at a higher rate.

A. ARQ/FEC Protocol with RCPC Codes

The ARQ/FEC protocol performs the following steps while transmitting a block I of information, which contains typically n = 100 to 1000 bits.

Encoding:

1) Add n_c parity bits to form an error detection code C_0, i.e., 16 or more CRC bits.

2) Add M "0" or known bits to properly terminate the encoder memory and the decoder trellis for the "information" block ($n + n_c$). M is the memory of the convolutional encoder.

3) Use the $1/N$ encoder to encode the convolutional code and store it at the transmitter, possibly in a matrix as in Fig. 8. The matrix is only conceptual and can be modified in several ways to accommodate interleaving.

Transmitting and decoding:

1) Transmit the first P columns of the matrix.

2) Transmit additional columns of the matrix up to index l according to the puncturing table.

3) Decode RCPC code of rate $P/(P + l)$ with Viterbi algorithm.

4) Check the syndrome of code C_0. If the syndrome is zero output I and send ACK to the transmitter. If the syndrome of C_0 is not zero increase l and repeat steps (2)–(4).

If decoding is still not successful for $l = (N - 1)P$, i.e., $R = 1/N$, several possibilities exists. A higher order protocol could take over or ask for a repetition of the whole procedure starting at step 1). Code combining as suggested by Chase [15] could be used to repeat the whole $1/N$ code L_r times and combine it with the already transmitted code. Actually, the VA uses in this case the same trellis as for all the RCPC codes, but the ML metric (3) is changed to

$$\lambda_j = \sum_{r=1}^{L_r} \sum_{i=1}^{N} a_{ij} a_{ijr} x_{ij} y_{ijr} = \sum_{i=1}^{N} a_{ij} x_{ij} \sum_{r=1}^{L_r} a_{ijr} y_{ijr} \qquad (15)$$

where index r indicates the rth repeated transmission. Note that by repeating all code bits L_r times the free distance is increased by a factor of L_r [15] and the numbers of the weight spectrum are unchanged

$$c_{d_r} = c_d \qquad d_r = L_r \cdot d \text{ for } d = d_{\text{free}} + i.$$

Note that a repetition code is almost optimal only for low rates such as 1/8, 1/16 [15]. By halving the code rate the distance usually is more than twice as shown in Table I for memory M = 4. This enables a gain in E_b/N_0 on the Gaussian channel. This type of code combining as expressed in (15) can be viewed as maximum ratio combining of the received y_{ijr} values. The combined y values which equal the inner sum of the right side in (15) are then used by the original code for ML decoding.

In the following analysis, we will assume that decoding stops with the $1/N$ code in step 4). Consequently, there exists

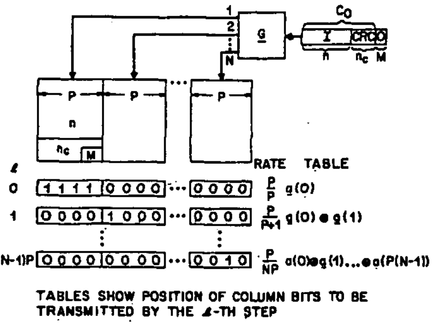

1
2
N
C_0
G
I
CRC
0
n
n_c
M
P
P
P
n
n_c
M
RATE TABLE
ℓ
0
1111 0000 ··· 0000
$\frac{P}{P}$ g(0)
1
0000 1000 ··· 0000
$\frac{P}{P+1}$ g(0) ● g(1)
(N-1)P
0000 0000 ··· 0010
$\frac{P}{NP}$ g(0)●g(1)...●g(P(N-1))
TABLES SHOW POSITION OF COLUMN BITS TO BE TRANSMITTED BY THE ℓ-TH STEP

Fig. 8. ARQ-FEC encoding scheme with RCPC codes.

a probability that the frame with information block I was not error-free decoded. This frame error rate FER will be calculated and should be sufficiently small. It should be pointed out that we are completely free in which steps we increase l. The number of steps depends on the ratio of frame arriving time (information rate) to frame decoding time (decoder speed). For extremely varying channel conditions an exponential increase could be useful, i.e., $l = 1, 2, 4, 8, 16$. This would mean to perform less FEC decoding attempts than the maximum number possible. This is at the cost of a slightly smaller throughput because in the example, the 10th attempt could have been successful, saving the transmission 11–16 in the above example.

With this type of protocol an acknowledgement (ACK) is only sent after successful decoding of a frame. The encoder would send more and more code bits until it receives the ACK. Assume that decoding is successful at step l_S. If there is a round-trip delay T_D involved in the transmission with rate $1/T_S$, the encoder would transmit T_D/T_S unnecessary bits, which would slightly reduce the throughput. This will be shown in the analysis. If the ACK is lost, the encoder keeps transmitting unnecessary code bits until $l = (N-1)P$. If the probability of an ACK loss or error is small the effect on the throughput is negligible. Thus, the return channel is used only once per frame and a bad return channel does not confuse the basic protocol.

B. Throughput Analysis

We will describe the performance of the hybrid ARQ/FEC scheme by two figures of merit: the average throughput R_{AV} and the probability that a frame or packet with n information bits is finally not correctly decoded, the so-called frame error rate FER. The throughput is

$$R_{AV}=\frac{n}{n+n_c+M}\cdot\frac{P}{P+l_{AV}}. \tag{16}$$

l_{AV} is the average number of additionally transmitted bits per P information bits. From Fig. 8 this can be interpreted as the average number of additionally transmitted columns. $P/(P+l_{AV})$ is the effective code rate of the FEC code. Due to the overhead of n_c parity check bits and M terminating bits with $n_c + M \ll n$, the throughput is slightly smaller than the effective FEC code rate. We assume that the error detecting code C_0 achieves a very low probability of undetected errors which will be ignored in the subsequent analysis. If the $(n + n_c, n)$ error detection code C_0 falls into the class which satisfies [20]

$$\text{Prob }\{\text{error undetected}\}<2^{-n_c}$$

a very low error probability can be achieved when n_c is at least 16. Typically, one would use a cyclic redundancy check CRC common in data transmission protocols. Let l_k be the sequence number of the decoding attempt using the RCPC code with puncturing tables $a(l_k)$ with

$$l_1<l_2<\cdots<l_k<\cdots l_K.$$

Let $P_{EF}(l_k)$ be the probability that the FEC decoding at the kth step results in errors which are detected by C_0. We will give an approximate analysis where we assume that the K decoding attempts have statistically independent outcomes. Then we have

$$l_{AV}=\sum_{k=1}^{K} l_k(1-P_{EF}(l_k))\cdot\prod_{i=0}^{k-1}P_{EF}(l_i)+l_K\prod_{i=0}^{K}P_{EF}(l_i) \tag{17}$$

where we define $P_{EF}(l_0) = 1$.

The frame error rate of the unsuccessful ARQ/FEC procedure is then

$$\text{FER}=P_{EF}(l_K). \tag{18}$$

We can include in the analysis a delay T_D caused by the round-trip time and the decoding time by replacing l_k in (17) by

$$l_k := \min\{l_K,\, l_k+PT_D/(T_S(n+n_c+m))\}. \tag{19}$$

Similarly, a loss or error in the ACK signal on the return channel, causing unnecessary transmission up to l_K, can be accommodated. For the probability $P_{EF}(l_k)$, we only can give an upper bound by using (9) to calculate the probability that an error event occurs for the transmitted bits

$$P_{EF}(l_k)<1-\left(1-\frac{1}{P}\sum_{d=d_{\text{free}}}^{\infty}a_d(l_k)\cdot P_d\right)^{n+n_c+M}. \tag{20}$$

The values a_d depend on the code, P_d depends on the channel and the detection scheme and can be selected from the values given in Section IV and in [9]. Recall that P_d usually assumes sufficient interleaving. By using (20) in (17) and (16), we obtain a lower bound for the throughput R_{AV} and by using (20) in (19), we get an upper bound on the FER.

Two examples will be given to show the performance of the RCPC codes in the hybrid ARQ/FEC scheme. The same code with $M = 4$, $N = 3$, $P = 8$ will be used, which has fairly low decoding complexity with 16 states and 32 add, compare and select operations per decoded bit. Fig. 9 shows the throughput on a Gaussian channel with hard and soft decisions as calculated by the lower bound method described. The simulations show that for this case the bound is accurate within 1 dB. The benefits of soft decisions are clearly observable. Due to the overhead R_{AV} approaches 0.97 for high SNR. At low SNR, R_{AV} approaches 0.323 and the frame error rate is high. The RCPC scheme which uses adaptive code rates between 1 and 1/3 has generally a better throughput than comparable schemes described in [4] and its throughput curve constitutes a kind of envelope to the corresponding fixed rate schemes. In Fig. 10 the performance on a fully interleaved Rayleigh channel is given. In this case, decoding starts with $l = 1$, i.e., the 8/9 code and ends with the 1/3 code. Several degrees of CSI are given as described in Section IV: YHAN, YHAH, and YSAS. Note that for a Rayleigh channel with $E_S/N_0 = 10$ dB a frame error rate of less than 10^{-6} and a throughput of 0.7 can be easily achieved. Simulation results and analytic bounds agree within 1 dB.

VI. The Application of RCPC Codes to Unequal Error Protection

In many applications it is required to provide different levels of error protection for different parts of an information sequence or block. Some examples have been given in the introduction. We assume that the source or the source coder supplies us with some information about the relative impor-

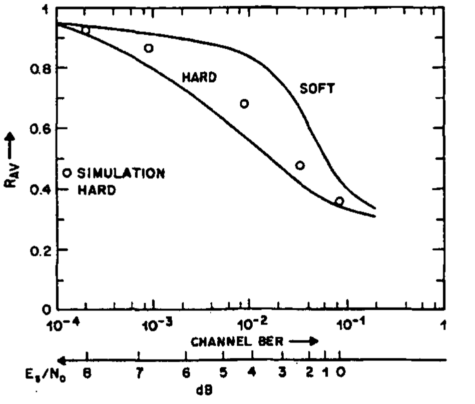


Fig. 9. ARQ/FEC throughput with RCPC codes on a Gaussian channel with soft and hard decisions (BSC). Code: $P = 8, N = 3, l = 0, 1, 2, 4, \cdots, 16, M = 4$. Frame: $n = 8 \times 125, n_c = 28$.

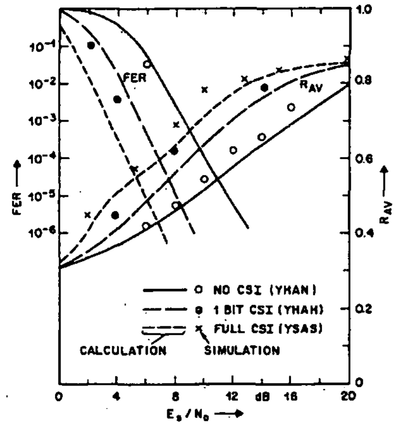
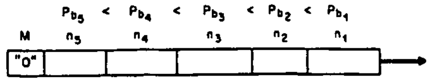


Fig. 10. ARQ/FEC throughput and frame error rate FER on an interleaved Rayleigh channel, hard decision, and CSI. Code: $P = 8, N = 3, l = 1, 2, 4, \cdots, 16, M = 4$. Frame: $n = 8 \times 48, n_c = 34$.

tance or susceptibility to errors of certain information bits or groups of information bits. We will call this source significance information (SSI). An example of an SSI would be the required BER after decoding for certain groups of information bits. Assume, that in a block of n information bits we have K groups with n_k information bits in the kth group requiring an BER of P_{bk} after decoding. In this case, P_{bk} would be the SSI and

$$\sum_{k=1}^{K} n_k = n.$$

An example is given in Fig. 11 where we have ordered the information bits according to their relative importance and their error protection needs P_{bk}. Of course one could now separately encode the K groups by K different encoders and use K different decoders according to the protection needs. Besides the increased overhead and complexity some numbers n_k might be rather small and exclude the use of powerful longer codes. Instead, we wish to use *one* encoder and *one* decoder which provides the protection requirements with a minimum of redundancy and overhead. Very little is known in the literature about unequal error protection of this type by block or convolutional codes. A recent reference is [21] where more literature is quoted. Most recently the extension of this concept to convolutional codes has been reported [26]. It seems, however, that only a limited amount of variability can be introduced and only short or low rate codes have been used. Furthermore, no ML decoding results have been shown. Another interesting unequal protection scheme with two codes has been reported in [22], which combines block and convolutional codes.



CODE INDEX l_k	l_5 >	l_4 >	l_3 >	l_2 >	l_1
CODE RATE R_k OF RCPC - CODE	$\frac{P}{P+l_5}$	< $\frac{P}{P+l_4}$	< $\frac{P}{P+l_3}$ <	$\frac{P}{P+l_2}$	< $\frac{P}{P+l_1}$
PUNCTURING TABLE $\underline{a}(l_k)$ OF RCPC-CODE	$a(l_5)$	$\underline{a}(l_4)$	$\underline{a}(l_3)$	$\underline{a}(l_2)$	$\underline{a}(l_1)$
FREE DISTANCE OF RCPC - CODE	d_5	> d_4	> d_3 >	d_2	> d_1

Fig. 11. Data frame with information bits grouped according to their error protection requirement (source significance information P_{bk}). RCPC codes assigned to the groups.

RCPC codes are well suited for this application. In the example of Fig. 11, the ordered information bits are shifted into the shift register of a $1/N$, memory M RCPC code. During the n_1 information bits the puncturing matrix $a(l_1)$ is used as the relevant rule for the multiplexer. As soon as the first bit of the second group enters the encoder the puncturing table, $a(l_2)$ will be used. After another n_2 information bits or encoder shifts, the table is switched to $a(l_3)$, etc. The procedure is easier to follow when n_k is an integer multiple of the puncturing period P, however, this is not a strict requirement. At the kth step $P + l_k$ code bits are transmitted per P information bits using the RCPC code with index l_k and rate $R_k = P/(P + l_k)$. The frame is terminated after the group n_K by shifting M "0" bits into the shift register, thus transmitting M/R_K overhead code bits necessary for proper termination of the trellis. The average code rate is then

$$R = \frac{\sum_{k=1}^{K} n_k}{\sum_{k=1}^{K} n_k \cdot (P + l_k)/P + M(P + l_K)/P}$$

The rate-compatibility condition (7a) is crucial. In a transitional phase between two matrices $a(l_k)$ and $a(l_{k+1})$ where $l_{k+1} > l_k$ we have to guarantee, that despite the transition, the distance properties of all paths originating in code l_k do not suffer a loss of distance due to transitions, thus guaranteeing at least the designed performance. Fig. 12 shows the same error paths relative to the correct all-zero path in RCPC-code l_k: path 1 having distance d_1 is not effected by a transition, whereas path 2 extends into the lower rate code l_{k+1}. The rate-compatibility condition (7a) guarantees that the l_{k+1} code does not puncture any "1" on the path. Due to less puncturing it might happen that additional "1" 's occur on the part B of the path, which would cause a $d_2 > d_1$. Therefore, some of the $\{a_d\}$ and $\{c_d\}$ values have to be assigned to higher distances if the path is not terminated at the transition. Thus, the rate compatibility restriction guarantees that transitional paths do have a distance which is at least the distance of the same path within the higher rate code and at most the distance of the same path within the lower rate code. If two punctured codes without the rate compatibility restriction are used at a rate transition, it can happen that a transitional path has a distance which is even lower than the distance of the same path

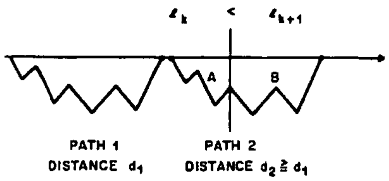


Fig. 12. Error paths in a time varying RCPC code during a transition period between code rates.

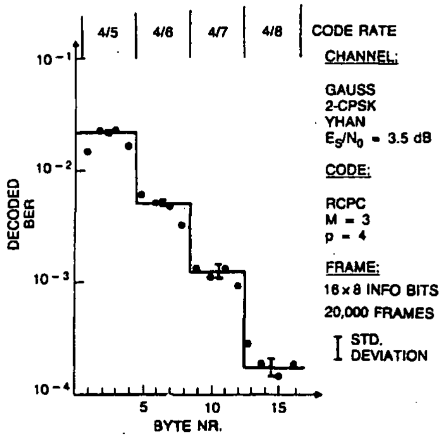


Fig. 13. Measured short term bit error rate in a data frame with unequal error protection using RCPC codes.

within the higher rate code. This would lead to a bad error behavior in the transition region. One can easily construct such an example.

It is useful to order the information as described with increasing importance. Decoding errors typically occur at the beginning of paths and the last M information bits of each error path are error free. However, the transitional paths (path 2) to lower rates generally have higher distances and therefore errors occur with lower probability. Simulations show that this type of ordering leads to a smaller number of total errors. Error paths can extend over many transitions. All paths will be properly terminated by the tail forcing the decoder to the all-zero state. Ordering the information bits from lower to higher rate codes would require proper zero termination at each change of rate. We conclude this section by giving an example of the simulated performance of the RCPC code with unequal error protection within one data frame of 128 bits. As Fig. 13 shows the code rate is changing from 4/5 to 4/8 while the measured short term BER is improving by a factor of 100. As discussed above, the bits close to a transition to the next lower rate get a better protection.

VII. Conclusions

We have described the concept of forward error correction codes which are obtained from low rate convolutional codes by puncturing them periodically in a rate-compatible manner. Surprisingly, this restriction still produced good codes relative to the best known codes of comparable complexity. In such a way, a whole family of codes with different rates is available using the same encoder and the same ML-decoder employing the Viterbi algorithm. Only the puncturing rule, i.e., the multiplexer switching pattern is changed. Since in most transmission schemes, channel rate is fixed due to modulation and channel filter requirements, changing the code rate means a change of information rate and a buffer for the incoming information stream is required. One disadvantage of punctured codes with high rate is that error paths can be quite long, as shown in Fig. 3 and in [7]. This requires either long path memories in the VA, framed transmission as employed in most packet oriented transmission schemes, or periodically inserted synchronization patterns or known data bits. In a decoder implementation, the basic Viterbi algorithm runs on the same trellis for all the codes. This suggests a VLSI for the Viterbi-decoding algorithm where only the control and the memory access is changed according to the rate, the channel conditions, and the channel state information (CSI).

Since the compatibility condition ensures that only incremental redundancy is used for lower rate codes, RCPC codes are suitable for ARQ/FEC protocols and unequal error protection as shown in Sections V and VI. Of course, both methods can be combined to adapt unequal error protection to varying channel conditions. For instance, the transmission security of a frame with a set of unequal error protection codes using rates (1, 8/10, 8/14) can be upgraded to a set of (8/10, 8/12, 8/16) codes when the channel conditions get worse by transmitting another two check bits per 8 information bits.

The concept of RCPC codes can be combined with the concept of nested codes [25] where the Viterbi decoder can work also with $M - 1, M - 2, \cdots$ on a reduced set of states. Codes have been found which satisfy both the RCPC and the nested condition. Furthermore, the concept of RCPC codes could also be used to find a family of "good" codes with high memory suitable for sequential decoding [3]. "Good" would mean that all the codes with different rates need a rapidly growing distance profile [3]. In an ARQ/FEC application, the error detection via the C_0 code could be replaced by a time-out condition in the Fano or stack algorithm.

The design of combined source and channel coding as described for voice transmission in [23] and for image transmission in [24] gets a new degree of freedom by using RCPC codes because the code rate can be changed within a data frame and during transmission according to the needs [27]. In other words, source significance information SSI, channel state information CSI, and check information from the decoded bits can be used for the adaptation of the encoding and ML-decoding algorithm of RCPC codes.

Acknowledgment

The author would like to thank H. Osthoff for writing the computer search program used in Section III as part of his Dipl. Ing. thesis at the Technical University München.

References

[1] G. C. Clark and J. B. Cain, *Error-Correction Coding for Digital Communications*. New York: Plenum, 1981.

[2] N. S. Jayant and P. Noll, *Digital Coding of Waveforms, Principles and Applications to Speech and Video*. Englewood Cliffs, NJ: Prentice Hall, 1984.

[3] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice Hall, 1983.

[4] S. Lin, D. J. Costello, and M. J. Miller, "Automatic repeat-request error control schemes," *IEEE Commun. Mag.*, vol. 12, pp. 5–17, Dec. 1984.

[5] D. M. Mandelbaum, "An adaptive-feedback coding scheme using incremental redundancy," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 388–389, May 1974.

[6] J. B. Cain, G. C. Clark, and J. M. Geist, "Punctured convolutional codes of rate $(n - 1)/n$ and simplified maximum likelihood decoding," *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 97–100, Jan. 1979.

[7] Y. Yasuda *et al.*, "Development of variable-rate Viterbi decoder and its performance characteristics," in *Proc. 6th Int. Conf. Digit. Satellite Commun.*, Phoenix, AZ, Sept. 1983, pp. XII-24–XII-31.

[8] Y. Yasuda, K. Kashiki, and Y. Hirata, "High rate punctured convolutional codes for soft decision Viterbi decoding," *IEEE Trans. Commun.*, vol. COM-32, pp. 315–319, Mar. 1984.

[9] J. Hagenauer, "Viterbi decoding of convolutional codes for fading and

burst channels," in *Proc. 1980 Int. Zurich Sem.*, Mar. 1980, pp. G2.1-G2.7.

[10] M. Mooser, "Some periodic convolutional codes better than any fixed code," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 750-751, Sept. 1983.

[11] M. Cedervall, "Contribution to the decoding and structure of convolutional codes," Ph.D. dissertation, Univ. Lund, Sweden, 1983.

[12] D. G. Daut, J. W. Modestino, and L. D. Wismer, "New short constraint length convolutional code constructions for selected rational rates," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 794-800, Sept. 1982.

[13] A. J. Viterbi and J. K. Omura, *Principles of Digital Communication and Coding*. New York: McGraw Hill, 1979.

[14] J. Hagenauer and E. Lutz, "Forward error correcting coding for fading compensation in mobile satellite channels," *IEEE J. Select. Areas Commun.*, vol. SAC-5, pp. 215-225, Feb. 1987.

[15] D. Chase, "Code combining—A maximum likelihood decoding approach for combining an arbitrary number of noisy packets," *IEEE Trans. Commun.*, vol. COM-33, pp. 385-393, May 1985.

[16] D. Chase, P. D. Mullins, and J. K. Wolf, "Application of code combining to a selective repeat ARQ link," in *1985 IEEE Military Commun. Conf. MILCOM '85, Conf. Rec.*, vol. 1, pp. 247-252, 1985.

[17] J. J. Metzner and D. Chang, "Efficient selective repeat ARQ strategies for very noisy and fluctuating channels," *IEEE Trans. Commun.*, vol. COM-33, pp. 409-416, May 1985.

[18] H. Krishna and S. D. Morgera, "A new error control scheme for hybrid ARQ systems," in *Proc. ICC 1986*, Toronto, Ont., Canada, pp. 637-645.

[19] T. Kasami, S. Lin, and D. J. Costello, "Performance analysis of a concatenated coding scheme for error control," in *Proc. ICC 1984*, Amsterdam, The Netherlands, pp. 769-773.

[20] J. K. Wolf, A. M. Michelson, and A. H. Levesque, "On the probability of undetected error for linear block codes," *IEEE Trans. Commun.*, vol. COM-30, pp. 317-324, Feb. 1982.

[21] W. J. Gils, "Linear unequal error protection codes from shorter codes," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 544-546, May 1984.

[22] L. Kittel and P. Piret, "Compound coding with unequal error protection for digital mobile radio systems," (in German), *NTG Fachberichte*, no. 90, "Bewegliche Funkdienste," pp. 178-185, VDE Verlag, Frankfurt, Germany, 1985.

[23] D. J. Goodman and C. E. Sundberg, "Combined source and channel coding for variable-bit-rate speech transmission," *Bell Syst. Tech. J.*, vol. 62, no. 7, pp. 2017-2036, Sept. 1983.

[24] D. G. Daut and J. W. Modestino, "Two dimensional DPCM image transmission over fading channels," *IEEE Trans. Commun.*, vol. COM-31, pp. 315-328, Mar. 1983.

[25] F. Hemmati, "A versatile Viterbi processor for decoding variable gain/rate convolutional codes," presented at Proc. 7th Int. Conf. Digital Satellite Commun., Munich, West Germany, May 1986.

[26] R. Palazzo, Jr., "On the linear unequal error protection convolutional codes," in *Proc. GLOBECOM 1986*, Houston, TX, pp. 1367-1371.

[27] R. V. Cox, J. Hagenauer, N. Senhadri, and C.-E. Sundberg, "A subband coder designed for combined source and channel coding," presented at Proc. ICASSP 1988, New York, Apr. 1988.

★

Joachim Hagenauer (M'78-SM'87) received the Ing.(grad) degree from Ohm-Polytechnic Nürnberg, in 1963, and the Dipl.-Ing. and the Dr.-Ing. degrees in electrical engineering from Technical University Darmstadt, in 1968 and 1974, respectively.

At Darmstadt University he served as an Assistant Professor and Dozent. From May 1975 to September 1976 he held a postdoctoral fellowship at the IBM T. J. Watson Research Center, Yorktown Heights, NY, working on error correction coding for magnetic recording. Since 1977 he has been with the German Aerospace Research (DFVLR) at Oberpfaffenhofen—since May 1980 as the Head of the Communication Theory Group. His research interests include convolutional coding, data transmission via fading channels, and mobile communications. During 1986/87 he spent a sabbatical year as an "Otto Lilienthal Fellow" at AT&T Bell Laboratories, Crawford Hill, NJ working on joint source/channel coding and on trellis coded modulation.